## IN THE CLAIMS

Please amend the claims as follows:

1. (Currently amended) A method ~~for supporting input/output for a virtual machine~~, ~~the method~~ comprising:

executing virtual machine application instructions of a virtual machine application by using micro architecture code of a processor architecture;

receiving an I/O access from the virtual machine application at a monitor supporting the virtual machine application;

~~upon~~ in response to receiving the I/O access, generating an exception;

performing the I/O access by using a host operating system configured to support the monitor;

updating state data for the virtual machine application at the monitor in accordance with the I/O access, the updating performed by a virtual machine component of the micro architecture code; and

resuming execution of the virtual machine application from the exception.

2. (Previously presented) The method of claim 1, wherein the micro architecture code comprises an instruction interpreter that functions with an

instruction translator to translate target instructions into host very long instruction word (VLIW) instructions to execute the virtual machine application instructions.

3. (Previously presented) The method of claim 1, wherein the micro architecture code comprises an instruction translator to execute the virtual machine application instructions.

4. (Previously presented) The method of claim 1, further comprising executing the monitor to implement the I/O access from the virtual machine application, wherein the monitor is configured to handle the exception caused by the I/O access.

5. (Currently amended) The method of claim 4, further comprising entering a single step mode, wherein the single step mode causes the monitor to single step through the virtual machine application instructions to handle the exception.

6. (Previously presented) The method of claim 4, further comprising using the monitor to maintain at least one virtual device to implement the I/O access from the virtual machine application.

7. (Previously presented) The method of claim 6, further comprising:

using the host operating system to access a real device in response to an access to the at least one virtual device; and

updating the state data for the virtual machine application in accordance with I/O data retrieved from the real device.

8. (Previously presented) The method of claim 1, wherein the virtual machine application instructions comprise target instructions, and the micro architecture code comprises host instructions.

9. (Previously presented) The method of claim 8, wherein the target instructions comprise x86 instructions, and the host instructions comprise VLIW instructions.

10. (Previously presented) The method of claim 8, wherein the virtual machine comprises an x86 compatible virtual machine.

11. (Currently amended) A system~~ for supporting input/output for a virtual machine~~, ~~the system ~~comprising,:

a processor architecture including micro architecture code ~~configured to~~that executes natively on a CPU hardware unit of the processor architecture; and a memory coupled to the processor architecture, the memory ~~storing~~ configured to store virtual machine application instructions of a virtual machine application, wherein the virtual machine application instructions are for execution using the micro architecture code to cause the system to implement a method comprising:

receiving an I/O access from the virtual machine application at a monitor supporting the virtual machine application;

~~upon~~ in response to receiving the I/O access, generating an exception;

performing the I/O access by using a host operating system configured to execute on the processor architecture and to support the monitor; and

updating state data for the virtual machine application at the monitor in accordance with the I/O access, the updating performed by a virtual machine component of the micro architecture code; and

resuming execution of the virtual machine application from the exception.

12. (Previously presented) The system of claim 11, wherein the micro architecture code comprises an instruction interpreter configured to function with an instruction translator to translate target instructions into host VLIW instructions to execute the virtual machine application instructions.

13. (Previously presented) The system of claim 11, wherein the micro architecture code comprises an instruction translator to execute the virtual machine application instructions.

14. (Previously presented) The system of claim 11, wherein the method further comprises executing the monitor to implement the I/O access from the virtual machine application, wherein the monitor is configured to handle the exception caused by the I/O access.

15. (Currently amended) The system of claim 14, wherein the method further comprises entering a single step mode which causes the monitor to single step through the virtual machine application instructions to handle the exception.

16. (Previously presented) The system of claim 14, wherein the method further comprises using the monitor to maintain at least one virtual device to implement the I/O access from the virtual machine application.

17. (Previously presented) The system of claim 16, wherein the method further comprises:

using the host operating system to access a real device in response to an access to the virtual device; and updating the state data for the virtual machine application in accordance with I/O data retrieved from the real device.

18. (Previously presented) The system of claim 11, wherein the virtual machine application instructions comprise target instructions, and the micro architecture code comprises host instructions.

19. (Previously presented) The system of claim 18, wherein the target instructions comprise x86 instructions, and the host instructions comprise VLIW instructions.

20. (Previously presented) The system of claim 18, wherein the virtual machine comprises an x86 compatible virtual machine.

21. (Currently amended) A computer readable medium having stored thereon, computer executable instructions that, if executed by a processor cause the processor to perform ~~A computer readable storage media for implementing support for an input/output process for a virtual machine, the storage media storing computer readable code which when executed by a processor causes the processor to implement~~ a method comprising:

executing virtual machine application instructions of a virtual machine application by using micro architecture code of a processor architecture;

receiving an I/O access from the virtual machine application at a monitor supporting the virtual machine application;

upon receiving the I/O access, generating an exception;

performing the I/O access by using a host operating system ~~configured to~~that executes on the processor architecture and ~~to~~ that supports the monitor;

updating state data for the virtual machine application at the monitor in accordance with the I/O access, the updating performed by a virtual machine component of the micro architecture code; and

resuming execution of the virtual machine application from the exception.

22. (Previously presented) The computer readable storage media of claim 21, wherein the micro architecture code comprises an instruction interpreter configured to function with an instruction translator to translate target instructions into host VLIW instructions to execute the virtual machine application instructions.

23. (Previously presented) The computer readable storage media of claim 21, wherein the micro architecture code comprises an instruction translator to execute the virtual machine application instructions.

24. (Previously presented) The computer readable storage media of claim 21, wherein the method further comprises executing the monitor to implement the I/O access from the virtual machine application, and wherein the monitor is configured to handle the exception caused by the I/O access.

25. (Currently amended) The computer readable storage media of claim 24, wherein the method further comprises entering a single step mode which causes the monitor to single step through the virtual machine application instructions to handle the exception.

26. (Previously presented) The computer readable storage media of claim 24, wherein the method further comprises using the monitor to maintain at least one virtual device to implement the I/O access from the virtual machine application.

27. (Previously presented) The computer readable storage media of claim 26, wherein the method further comprises:

using the host operating system to access a real device in response to an access to the virtual device; and

updating the state data for the virtual machine application in accordance with I/O data retrieved from the real device.

28. (Previously presented) The computer readable storage media of claim 21, wherein the virtual machine application instructions comprise target instructions, and the micro architecture code comprises host instructions.

29. (Previously presented) The computer readable storage media of claim 28, wherein the target instructions comprise x86 instructions, and the host instructions comprise VLIW instructions.

30. (Previously presented) The computer readable storage media of claim 28, wherein the virtual machine comprises an x86 compatible virtual machine.

31. (cancelled)

32. (cancelled)